基于 OAuth2.0 的认证授权技术①

时子庆[1]，刘金兰[1]，谭晓华[2]

[1](天津大学 管理与经济学部，天津 300072)
[2](天津大学 机械学院，天津 300072)

摘 要：开放平台的核心问题是用户验证和授权问题，OAuth 是目前国际通用的授权方式，它的特点是不需要用户在第三方应用输入用户名及密码，就可以申请访问该用户的受保护资源。OAuth 最新版本是 OAuth2.0，其认证与授权的流程更简单、更安全。研究了 OAuth2.0 的工作原理，分析了刷新访问令牌的工作流程，并给出了 OAuth2.0 服务器端的设计方案和具体的应用实例。

关键词：开放平台；开放 API；用户认证和授权；OAuth2.0 协议；OAuth Server

Authentication and Authorization Technique Based on OAuth2.0

SHI Zi-Qing[1], LIU Jin-Lan[1], TAN Xiao-Hua[2]

[1](School of Management, Tianjin University, Tianjin 300072, China)
[2](School of Mechanical Engineering, Tianjin University, Tianjin 300072, China)

Abstract: The essential problem of open platform is the validation and authorization of users. Nowadays, OAuth is the international authorization method. Its characteristic is that users could apply to visit their protected resources without the need to enter their names and passwords in the third application. The latest version of OAuth is OAuth2.0 and its operation of validation and authorization are simpler and safer. This paper investigates the principle of OAuth2.0, analyzes the procedure of Refresh Token and offers a design proposal of OAuth2.0 server and specific application examples.

Key words: open platform; open API; authentication and authorization; OAuth2.0 protocol; OAuth server

1 引言

传统互联网时代，各个网站和服务之间是封闭的，数据无法进行交互。随着互联网技术的迅猛发展、系统之间的相互协作日益增多，开放与共享数据的需求不断增加，互联网服务之间的整合已经成为必然趋势，一个通过自身开放平台来实现数据互通甚至用户共享的时代已经来临。把网站的服务封装成一系列计算机易识别的数据接口开放出去，供第三方开发者使用，这种行为就叫做 Open API，提供开放 API 的平台本身就被称为开放平台。

2007 年 5 月份，Facebook 宣布改版，最早提出了开放平台的概念，正式从一个社交网站向一个社交应用平台转型。至此，开放平台发展迅速，成为互联网发展的趋势，也是一种革命性的发展模式。

国内外已经有很多的公司开发了自己的开放平台，Facebook、Twitter、腾讯、新浪微博、人人网，它们用户规模大、技术实力强，为第三方提供了一整套自成体系、纷繁复杂的“开放 API”。通过开放平台，网站不仅能提供对 Web 网页的简单访问，还可以进行复杂的数据交互，允许第三方开发者利用其资源开发复杂的应用，既丰富自身网站应用，为用户提供更好的服务，逐步建立起一个服务完备的网络社会，也为第三方连接的网站带来更多的用户。开放平台迅速成为互联网发展的趋势。

开放平台的核心问题在于用户验证和授权。对于服务提供商来说，一般不会希望第三方直接使用用户

① 收稿时间:2011-07-13;收到修改稿时间:2011-08-21

名和密码来验证用户身份，除非双方具有很强的信任关系。OAuth 协议正是为了解决服务整合时“验证和授权”这一根本问题而产生的。

OAuth 协议为用户资源的授权提供了一个安全的、开放而又简易的标准。与以往的授权方式不同之处是 OAuth 的授权不会使第三方触及到用户的用户名与密码等敏感信息，即第三方无需使用用户的用户名与密码就可以申请获得该用户资源的授权[3]。

OAuth 最新版本是 OAuth2.0，其认证与授权的流程更简单、更安全。本文研究了 OAuth2.0 的工作原理，分析了刷新访问令牌的工作流程，并给出了 OAuth2.0 服务器端的设计方案。

2 OAuth2.0认证授权技术

2.1 OAuth 协议

OAuth 1.0 已经在 IETF 尘埃落定，编号是 RFC5894。标志着 OAuth 正式成为互联网标准协议。

目前 OAuth 2.0 还没有最后定稿，从 2010 年 4 月到 2011 年 6 月，该协议已经发展出 16 个子版本。OAuth 2.0 的关注点在于简化客户端程序员的工作，以简化实现为原则，并对更多的接入形式予以支持，如按照官方的描述，OAuth 2.0 能够同时支持“Web 应用、桌面应用、移动终端、家庭设备”等等。OAuth 2.0 是一套全新的协议，相信经过多年的发展，OAuth 2.0 将成为未来开放平台领域标准的授权协议，并且随着技术发展，这将不仅仅是一个简单的协议，而会成为一个解决各种环境下授权问题的标准的协议族。

2.2 OAuth2.0

此文所介绍的 OAuth 的工作流程和服务器端的实现等，都是基于 OAuth2.0 协议的。OAuth 2.0 协议的新特性主要表现在以下几个方面：

(1) 授权许可的方式多样化

OAuth2.0 为了适应各种授权环境，把多种客户端流程和多种授权许可的方式结合起来，归结为以下几种获取授权的方法：

Authorization Code：客户端是 web 服务器程序的一部分，通过 http 请求实现，是 OAuth 1.0 流程的简化版本；

Implicit Grant：客户端运行于用户代理内（通常用 JavaScript 等脚本语言在浏览器中实现）；

Resource Owner Password Credentials：这种授权许可的类型需要最终用户和客户端有很强的信任关系，客户端可以直接使用资源拥有者的私有证书（用户名和密码）用作访问许可来获取 Access Token；

Client Credentials：客户端使用它的私有证书（client_id 和 client_secret）去获取 Access Token。

(2) 无需加密的认证方式

OAuth 2.0 提供一种无需加密的认证方式，此方式是基于现存的 cookie 验证架构，Access Token 本身将自己作为 secret，通过 HTTPS 发送，从而替换了通过 HMAC 和 token secret 加密并发送的方式，既简单，又安全。OAuth2.0 也对签名机制进行了也大大的简化。

(3) 用 Refresh Token 来刷新 Access Token

在 OAuth1.0 中，授权服务器会发行一个有效期非常长的 Access Token(典型的是一年有效期或者无有效期限制)。在 OAuth 2.0 中，授权服务器将发行一个短有效期的 Access Token 和长生命期的 Refresh Token。这将允许客户端无需使用用户再次操作而获取一个新的 Access Token，并且也限制了 Access Token 的有效期。

3 OAuth2.0的工作流程

3.1 重要术语

Authorization Server：授权服务器，能够成功验证资源拥有者和获取授权，并在此之后分发令牌的服务器；

Resource Server：资源服务器，存储用户的数据资源，能够接受和响应受保护资源请求的服务器；

Client：客户端，获取授权和发送受保护资源请求的第三方应用；

Resource Owner：资源拥有者，能够对受保护资源进行访问许可控制的实体；

Protected Resource：受保护资源，能够使用 OAuth 请求获取的访问限制性资源；

Authorization Code：授权码；

Refresh Token：刷新令牌；

Access Token：访问令牌。

3.2 OAuth2.0 协议的核心工作流程

OAuth 为客户端提供了一种代表资源拥有者访问受保护资源的方法。在客户端访问受保护资源之前，它必须先从资源拥有者获取授权（访问许可），然后用访问许可交换访问令牌（Access Token，包含许可的作用域、持续时间和其它属性等信息）。客户端通过向资

源服务器出示访问令牌来访问受保护资源。

访问令牌提供了一个抽象层，将不同的授权结构（如用户名密码）替换成资源服务器可以理解的单一访问令牌。这种抽象使得分发短期有效的访问令牌成为可能，也使得资源服务器不必理解多种多样的授权机制。

使用 OAuth2.0 机制，进行认证授权，获取访问令牌，并通过访问令牌来访问受保护资源，如图 1 所示：

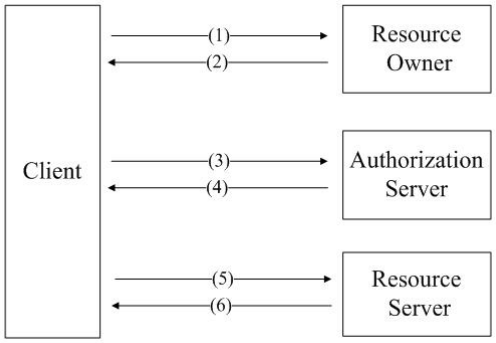

图 1　OAuth2.0 工作流程示意图[2]

OAuth2.0 的工作流程简述如下：

(1) 客户端从资源拥有者（最终用户）那里请求授权。授权请求能够直接发送给资源拥有者，或者间接的通过授权服务器发送请求；

(2) 资源拥有者为客户端授权，给客户端发送一个访问许可(Authorization Code)；

(3) 客户端出示自己的私有证书(client_id 和 client_secret)和上一步拿到的访问许可，来向授权服务器请求一个访问令牌；

(4) 授权服务器验证客户端的私有证书和访问许可的有效性，如果验证有效，则向客户端发送一个访问令牌，访问令牌包括许可的作用域、有效时间和一些其他属性信息；

(5) 客户端出示访问令牌向资源服务器请求受保护资源；

(6) 资源服务器对访问令牌做出响应。

3.3 OAuth2.0 刷新访问令牌的方式

OAuth 最开始的版本，是为第三方发放一个有效期非常长的访问令牌(Access Token)，有效期是一年，甚至是无期限限制。如果这样的话，有很大的安全隐患。首先，第三方拿到这个访问令牌，如果是无限期的话，第三方可以随时拿着访问令牌去访问资源拥有者的受保护资源，甚至是修改受保护资源；其次，如果这个访问令牌和访问令牌密钥被其他人拿到的话，就更没有安全性可言了。

在 OAuth2.0 中，授权服务器将发送一个短期有效的访问令牌(Access Token，一般有效期是 24 小时)和一个长有效期的 Refresh Token。当 Access Token 过期之后，第三方出示其私有证书(client_id 和 client_secret)和该用户的 Refresh Token，重新获取有效的 Access Token 和 Refresh Token（可选），此时，Refresh Token 担当起了访问许可的角色。这样，既限制了 Access Token 的有效期，也无需资源拥有者再次操作去获取一个新的 Access Token[2]。

OAuth2.0 刷新访问令牌的工作流程示意图，如图 2 所示：

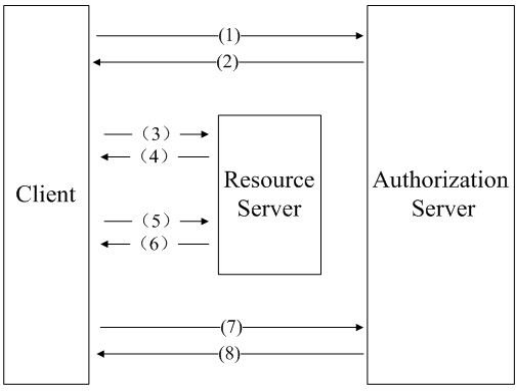

图 2　刷新访问令牌[2]

OAuth2.0 刷新访问令牌的工作流程简述：

(1) 客户端向授权服务器出示自己的私有证书(client_id 和 client_secret)和资源拥有者给它的访问许可；

(2) 授权服务器验证客户端私有证书和访问许可的有效性，并分发一个访问令牌和刷新令牌；

(3) 客户端出示访问令牌向资源服务器请求受保护的资源；

(4) 资源服务器验证访问令牌的有效性，并响应这个请求；

(5) 客户端不断通过访问令牌向资源服务器请求受保护资源，直到访问令牌过期；

(6) 访问令牌失效，资源服务器返回一个错误信息；

(7) 客户端出示它的私有证书和 Refresh Token，来请求一个新的访问令牌；

(8) 授权服务器，验证客户端私有证书和 Refresh Token 的有效性，如果有效，则分发一个新的访问令牌（也可以要求再分发一个刷新令牌）。

4 OAuth Server 的设计

虽然 OAuth2.0 还没有最后定稿，但是，已经有不少大型互联网公司宣布支持 OAuth2.0 协议，并且，现在已经有公司实现了 OAuth2.0 机制的验证和授权。

OAuth 2.0 提供一种无需加密的认证方式，使认证过程更加简单、安全，但是，验证信息必须要通过 HTTPS 发送，防止被恶意截取，因此，需要购买 SSL 证书，并在服务器端增加 SSL 模块，配置 HTTPS 服务。

OAuth2.0 服务器端的设计，本文只对 OAuth2.0 最核心的工作流程进行设计，即获取 Access Token 的过程。

4.1 服务器端的数据存储方案

OAuth2.0 机制的核心工作流程，大部分信息，如 APP 信息、用户和 APP 之间的授权关系、Access Token 等，需要永久性存储，存储在数据库中。一些不需要永久存储的信息，如临时授权码，只使用一次，而且，在很短的时间内就要使其失效，就没有必要存储在数据库之中，可存储在譬如 memcached 等缓存系统中，即提高了系统的处理速度，又减少了数据库压力。

数据库表结构设计：

客户端要先注册一个应用，获取该应用的 APPID 和 APPSECRET，应用的详细信息存储在数据表中，如表 1 所示：

表 1 appinfo APP 信息存储表

字段	备注
appid	client_id
appsecret	client_secret
appname	应用名称
appowner	应用的所有者
owneremail	应用拥有者的 email
appdescribe	应用描述
status	应用是否通过审核
callbackurl	跳回的 url
addtime	添加时间
……	其他

客户端与资源拥有者是“多对多”的关系，如下表所示：

表 2 authorize 授权关系存储表

字段	备注
appid	应用 ID
userid	用户 ID
addtime	添加时间
……	其他

表 3 access_token 访问令牌存储表

字段	备注
access_token	访问令牌
addtime	添加时间
……	其他

Memcached 存储访问许可

使用 memcached 缓存系统存储临时授权码，既提高系统的反应速度，又减少数据库的访问压力。

Key：{authorization_code}

Value：

array{

appid=>client_id

generatetime=>授权码产生的时间

isused=>是否已经使用过

}

4.2 服务器端需要提供的接口和页面

服务器端需要提供两个接口和两个页面：两个接口，一个是用来获取访问许可的接口，一个是用来获取 Access Token 的接口；然后再提供一个登录页面和一个授权页面。

4.2.1 获取访问许可（Authorization Code）接口

客户端从资源拥有者（最终用户）那里请求授权。授权请求直接发送给资源拥有者，如果资源拥有者为该客户端授权，则给客户端发送一个访问许可(Authorization Code)：

传入参数：

appid 应用 ID

callback_url 回调 URL

返回信息：

未登录、未授权：跳转到登录页面；

已经登录、未授权：跳转到授权页面；

已经登录、已经授权：跳转到传入的 callback_url 页面，并返回一个访问许可。

4.2.2 登录页面

这个页面展示给资源拥有者，输入用户名、密码，

登录成功之后，如果未授权，则跳转到授权页面；如果已经授权，直接跳转到 callback_url 页面，并返回一个访问许可。

4.2.3 授权页面

这个页面也是展示给资源拥有者，如果用户已经授权：则直接跳转到 callback_url 页面，并返回一个访问许可；如果用户未授权：出现授权页面，授权成功之后，跳到 callback_url 页面，并返回一个访问许可。

4.2.4 获取 Access Token 接口

客户端向授权服务器出示自己的私有证书和上一步拿到的访问许可，来请求一个访问令牌（Access Token）；授权服务器验证客户端的私有证书和访问许可的有效性，如果验证有效，则向客户端发送一个访问令牌，访问令牌包括许可的作用域、有效时间和一些其他属性信息：

传入参数：

Appid 应用 ID，client_id

Appsecret client_secret

callback_url 回调 URL

authorization_code 访问许可

返回信息：

Access Token，Access Token 中包含 APPID 和用户信息 userID 等。

获得 Access Token 之后，就可以向资源服务器出示 Access Token，访问受保护资源。

5 OAuth2.0 的具体应用

OAuth2.0 协议，解决了在开放平台中，服务整合时用户的“验证和授权”问题。百度开放平台、新浪微博开放平台、腾讯开放平台的用户验证和授权都是用的 OAuth1.0 协议，但是 OAuth1.0 只有单一的认证流程，认证过程复杂，用户体验差。

2010 年 4 月，OAuth2.0 协议的第一个草稿出来之后，Facebook 和 Twitter 甚至已经宣布在官方发布第一个草案前的就会开始支持 OAuth 2.0。虽然 OAuth2.0 协议还没有最后定稿，但由于它的种种优点，事实上，已经有支持 OAuth2.0 的产品了，服务器端包括 facebook 的 Graph Api，客户端方面，苹果的 iPhone 和 iPad 上也已经有具体实现了。

2011 年 1 月，人人网开放平台 OAuth2.0 验证授权服务正式上线。人人网开放平台是国内第一家实现 OAuth2.0 验证授权协议的平台。目前，人人网开放平台的 OAuth2.0 能够同时支持“Web 应用、桌面应用、移动终端、家庭设备”等等，并且支持更加安全的 HTTPS 协议。

为了更好的研究 OAuth2.0 认证授权技术，我们在人人网开放平台创建了一个应用，人人网暗恋系统“悄悄喜欢你”(qiaoqiaoai.com)，利用人人网开放平台的 OAuth2.0 授权认证技术进行认证，获得 Access Token 之后，利用 Access Token 和其他一些相关参数，调用人人网的开放 API，来获取用户的详细信息和好友列表信息等，完成暗恋系统的功能。目前，在人人网暗恋系统中，通过 OAuth2.0 认证授权的用户已经迅速超过 50 万人，可见开放平台不仅会对一些应用或者网站系统带来用户数量的激增，也会推动互联网向更加开放的模式改革。

6 结语

OAuth 是目前国际通用的授权方式，它的特点是不需要用户在第三方应用输入用户名及密码。其最新版本 OAuth2.0，认证与授权的流程更简单、更安全。本文详细介绍了 OAuth2.0 认证与授权技术的工作流程，而且也对其服务器端的实现给出了一个具体方案。虽然 OAuth2.0 还没有最后定稿，但相信，在不久的将来，OAuth 2.0 将成为未来开放平台领域标准的授权协议，并且随着技术发展，这将不仅仅是一个简单的协议，而会成为一个解决各种环境下授权问题的标准协议族。

参考文献

1 González JF, Rodríguez MC, Nistal ML, et al. Reverse OAuth: A solution to achieve deleated authorizations in single sign-on e-learning systems. Computers & Security, 2009,28:43–856.

2 The OAuth 2.0 Authorization Protocol (draft–ietf-oauth–v2–16),http://tools.ietf.org/html/draft-ietf-oauth-v2-16,2011.5

3 The OAuth 1.0 Protocol. http://tools.ietf.org/html/rfc5849, 2010.4.

4 张卫全,胡志远.浅析作用于 Web2.0 安全防范的 OpenID 和 OAuth 机制.通信管理与技术,2011,4(2):15–18.

5 许彤,雷体南.OpenID 与 OAuth 技术组合应用于教学资源库建设.软件导刊(教育技术),2009,10:69–70.